Proceedings of the 2007 American Control Conference
Marriott Marquis Hotel at Times Square
New York City, USA, July 11-13, 2007

Analysis and experimental validation of a sensor-based event-driven controller

J.H. Sandee, W.P.M.H. Heemels, S.B.F. Hulsenboom and P.P.J. van den Bosch

Abstract— **We present an event-driven servo controller that is based on an (extremely) low resolution encoder. The control value is updated at each moment that an encoder pulse is detected, yielding zero measurement error. However, as the time between two control updates is varying now, conventional controller design methods do not apply as they normally assume a constant sample time. To deal with this problem, the controller design is performed by transforming the system equations from the time domain to the position (spatial) domain, in which the encoder pulses, and therefore the controller triggering, are equidistant. In this way, the control problem is rewritten as a synchronous problem for a non-linear plant. A gain scheduled controller is designed and analyzed in the spatial domain. This event-driven controller is experimentally validated on a prototype printer where a 1 pulse per revolution encoder is used to accurately control the motion of images through the printer.**

I. INTRODUCTION

In industry we observe an ever increasing search for better performing products at decreasing cost prices, especially for consumer products that are sold in large quantities. Although cost price should decrease, the requirements become more challenging. In the document printing industry the same trend is observed. Printers should operate at higher printing speeds, be able to handle more media simultaneously and produce more accurate prints for the same or even a lower cost price. On top of that, the demands for power consumption and machine size are tightened. Because of these challenging requirements and the need to reduce the cost price, trade-offs appear for many aspects in the product design and system designers are forced to come up with creative solutions for these hard problems.

One of these challenging problems in the design of a printer, but also in many other high-tech systems, is the servo control of several motors at high accuracy. Because of cost price requirements conventional solutions are often not feasible anymore. High resolution encoders are too expensive and high sample frequencies are also prohibitive as controllers have to run on low-cost processors with processing power that is shared with many other tasks.

One of the leading companies in high volume document printing systems is Océ Technologies BV. The technology that this company has developed for high speed toner printing is the 'Copy Press system' (see figure 1). In this technique the toner of the image is transported from the masterbelt to the fuse roll (where the image is fused onto the sheet) with the Toner Transfer Fusing (TTF) technology. This technology uses a special belt (TTF belt) that transports the image. This TTF belt is accurately controlled by a brushless DC-motor. Accurate control is important as the positioning accuracy of the image directly influences the printing quality.

The brushless DC-motor includes three Hall sensors (sensors that can detect magnets connected to the rotor, and therefore the position of the rotor) to implement correct commutation, as no brushes are available, like in the more commonly used DC-motors. Because of the omission of brushes, brushless DC-motors have a large lifetime and high reliability. In controlling the motor, conventional control algorithms use expensive, high resolution encoders to accurately measure the angular position. For instance, in the conventional printer setup, typical encoder resolutions are 500 pulses per revolution (PPR), with controllers running at 500 Hz. By using these high resolution encoders, measurement quantization errors can be neglected.

To keep the system cost price limited, our aim is to use only the Hall sensors to control the motor. The obtainable resolution is thereby limited to 12 PPR in the practical case-study (after demodulation). However, now the quantization errors become significant and are not negligible anymore. To still achieve satisfactory control performance, this requires an adaption to conventional control algorithms to deal with this low-resolution encoder signal.

Most applied and researched solutions that deal with noisy and low resolution sensor data use an observer-based approach to estimate the data at *synchronous* controller sample moments, based on *asynchronous* measurement moments [3], [5], [7], [8], [13], [14], [15]. In these solutions, the

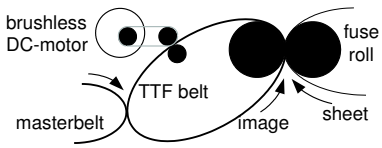

Fig. 1. Schematic representation of the 'Copy Press system'.

[1]This work has been carried out as part of the Boderc project under the responsibility of the Embedded Systems Institute. This project is partially supported by the Dutch Ministry of Economic Affairs under the Senter TS program.

Heico Sandee and Paul van den Bosch are with the Eindhoven University of Technology, Dept. of Electrical Engineering, Control Systems Group, P.O. Box 513, 5600 MB Eindhoven, The Netherlands. Heico Sandee is also with Océ Technologies BV. j.h.sandee@tue.nl, p.p.j.v.d.bosch@tue.nl

Maurice Heemels is with the Eindhoven University of Technology, Dept. of Mechanical Engineering, Control Systems Technology Group, P.O. Box 513, 5600 MB Eindhoven, The Netherlands. m.heemels@tue.nl. Maurice is also with the Embedded Systems Institute (ESI), The Netherlands.

Sander Hulsenboom is with Océ Technologies BV, PO Box 101, 5900 MA Venlo, The Netherlands. sander.hulsenboom@oce.com

continuous-time plant is translated into a discrete-time model which is time-varying, as it depends on the time between successive measurement instants. The approaches in [8], [13] use Kalman filtering [10], while in [14], a Luenberger-type observer is applied to use asynchronous measurement data in combination with a multi-rate controller scheme. In tracking applications, a well-known technique is the $\alpha\beta$-tracker [3], [7] to estimate position, velocity and acceleration in a time-discrete manner, often based on extrapolating and filtering measurements. An overview of these methods and a comparison between them is presented in [15] in the application of using optical incremental encoders to measure position and velocity. All these methods have proven to achieve a better control performance compared to the situation in which one neglects the quantization and sensor noise in the measurements. However, the main drawback of these methods is that they generally require a high computational effort for computing the observer estimates. Furthermore, next to the control parameters, also the observer parameters need to be tuned to get good overall control performance.

In this paper we will use a completely different control paradigm that is simple to implement and does nog suffer from the added complexity of an observer. The control structure is an asynchronous control scheme in which the control updates are triggered by the position measurement (encoder pulse). The idea of the asynchronous controller is based upon the observation that at an encoder pulse the position is *exactly* known and thus there is no need for an observer of filter as in the before mentioned approaches. However, as the velocity of the motor varies over time, both measurement and control updates are not equidistant in time. This requires a completely new design strategy for these event-driven controllers of which initial proposals are made in [11]. This paper applies this event-driven controller and extends the controller analysis and design techniques to accurately control the brushless DC-motor in the printer on the basis of a very low resolution Hall encoder. The method to design and tune the controller is presented. Furthermore, simulation and experimental results are compared with an $\alpha\beta$-tracker approach, as originally applied to control the motion of the TTF belt. This latter approach we call the observer-based controller. For the experiments we used the industrial setting of a high speed printer.

From a broader perspective, this work can be related to [1], [2] in which also event-driven feedback controllers are proposed as alternatives to conventional time-driven feedbacks. However, in [1], [2] typically the events are introduced in the system by design to reduce resource utilization, while here the events are intrinsic to the measurement device. Recently, [12] also considered event-driven control based on encoder pulses and uses quantized state controllers and deadbeat type of controllers. The work here uses a different type of controller (gain scheduled PD, triggered by encoder pulses) to achieve good reference tracking.

The outline of this paper is as follows. The problem statement is given in section II. Next, sections III and IV present two solutions for the problem: The conventional industrial solution in the form of a observer-based controller based on an $\alpha\beta$-tracker, and an event-driven controller, respectively. The design methods for the event-driven controller are presented in section IV as well. Simulation and experimental results are presented in sections V and VI, respectively. The paper ends with conclusions.

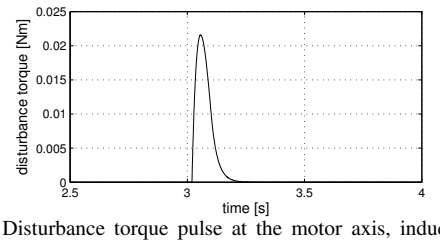

Fig. 2. Disturbance torque pulse at the motor axis, induced by a sheet entering the fuse roll.

II. PROBLEM FORMULATION

The brushless DC-motor that is driving the TTF-belt (figure 1) is modeled by the second-order model

$$\begin{aligned} \dot{\theta}(t) &= \omega(t) \\ \dot{\omega}(t) &= \tfrac{1}{J}[(\tfrac{-k^2}{R} - B)\omega(t) + \tfrac{k}{R}u(t) - d(t)] \end{aligned} , \qquad (1)$$

where $\theta(t)$ $[rad]$ is the angular position of the motor axis, ω(t) is its angular velocity $[rad/s]$, $u(t)$ the motor voltage $[V]$ and $d(t)$ the disturbance torque $[Nm]$ at time $t \in \mathbb{R}$. The motor parameters are obtained from data sheets of the manufacturer: the motor inertia $J_m = 0.83 \cdot 10^{-4}\ kgm^2$, the motor torque constant $k = 0.028\ Nm/A$, the motor resistance $R = 1.0\ \Omega$ and the motor damping $B = 3.0 \cdot 10^{-5}$ Nms/rad. The inertia of the load $J_l = 1.0 \cdot 10^{-4}\ kgm^2$ is added to the motor inertia to obtain the total inertia J.

The industrial requirements for throughput and printing accuracy in the printer result in a feedback and feed-forward controller combination such that:

- The deviation from the steady-state position error is at most 0.25 rad during printing. Only deviations from a constant position error (i.e. from a constant velocity) will be visible in the print quality.
- Position profiles are tracked corresponding to constant velocities ranging from 200 rad/s to 500 rad/s.
- Disturbances are rejected sufficiently up to frequencies of at least 4.0 Hz. This is defined as the required controller bandwidth. Below the derivation of this bandwidth is explained in more detail.

The main disturbances are caused by sheets that enter the fuse roll. The main component of this disturbance is due to the torque needed to open the fuse roll such that the sheet can enter. From measurements at the motor axis that drives the TTF-belt, the disturbance signal as induced by one sheet entrance in the fuse was obtained, as depicted in figure 2.

The requirement on the controller bandwidth is derived from the disturbance with the highest frequency that the controller needs to compensate. This disturbance is caused by the roll that is coupled to the motor and drives the TTF belt. As the gear ratio between motor and roll is 20:1, the main component of this disturbance has a period of $20 \cdot 2\pi$

rad at the motor axis. Because the disturbance is varying with the angular position of the motor, the frequency content of this signal varies with the velocity at which the motor is operated. At maximum printing speed of 100 pages per minute, which corresponds to a motor velocity of 500 rad/s, a disturbance at $\frac{500}{20 \cdot 2\pi} = 4.0$ Hz is observed. This disturbance can be modeled as a disturbing torque. From this the above mentioned requirement was derived that the controller has to sufficiently suppress disturbances up to at least 4.0 Hz.

To measure the angular position of the motor axis, the Hall pulses are used as an encoder with a resolution of 12 PPR. It is important to note that the Hall sensors are positioned along the motor axes with an inaccuracy of ± 0.2 rad (which is equal to 3% of one revolution).

III. OBSERVER-BASED CONTROL

In the observer-based control scheme, as originally implemented to control the motion of the TTF-belt, the actuator signal is updated at a constant rate with sample period T_s (i.e. synchronous in time) but measurements are done asynchronously in time. Each moment a new Hall pulse is detected, a time-stamp (τ_m) is taken. At each synchronous control update time (kT_s), this time-stamp is used to estimate ω and θ at the control update times kT_s from the asynchronous measurements. The estimates are denoted by ω_{est} and θ_{est}, respectively. This is done in two steps:

- Based on $\omega_{est}((k-1)T_s)$ (computed at the previous control update time $(k-1)T_s$) the latest measured angular position $\theta_m(\tau_m)$ is translated into an extrapolated angular position (θ_{extr}) at the next synchronous control update time kT_s. This is done by linear extrapolation at the discrete-time instants $k = 0, 1, 2, ...$:

$$\theta_{extr}(kT_s) = \theta_m(\tau_m) + (kT_s - \tau_m)\omega_{est}((k-1)T_s), \tag{2}$$

  where $(k-1)T_s \leq \tau_m < kT_s$.
- To deal with high frequent measurement noise, caused by the inaccurate positioning of the Hall sensors, the $\alpha\beta - tracker$ structure [7] was adopted to compute $\theta_{est}(kT_s)$ from $\theta_{extr}(kT_s)$. At the same time, the $\alpha\beta - tracker$ also computes $\omega_{est}(kT_s)$. The $\alpha\beta - tracker$ can be written as a weighted sum of the estimated values, based on information at time kT_s, and the "innovation terms", as derived from the extrapolated measurements:

$$\begin{array}{rcl} \theta_{est}(kT_s) & = & (1-\alpha)\{\theta_{est}((k-1)T_s) + \\ & + & T_s\omega_{est}((k-1)T_s)\} + \alpha\theta_{extr}(kT_s) \\ \omega_{est}(kT_s) & = & (1-\beta)\omega_{est}((k-1)T_s) + \\ & + & \beta\frac{\theta_{extr}(kT_s) - \theta_{est}((k-1)T_s)}{T_s} \end{array}, \tag{3}$$

  for some $0 < \alpha \leq 1$, $0 < \beta \leq 1$.

Based on the difference between the reference values for θ and ω (denoted by θ_r and ω_r, respectively) the error signals can be calculated that enter the controller:

$$\begin{array}{l} e_\theta(kT_s) = \theta_r(kT_s) - \theta_{est}(kT_s) \\ e_\omega(kT_s) = \omega_r(kT_s) - \omega_{est}(kT_s) \end{array} \tag{4}$$

To control the motor, a PD controller with feed-forward was used with the following structure:

$$u(t) = K_p e_\theta(kT_s) + K_d e_\omega(kT_s) + K_{ff}\omega_r(kT_s) \tag{5}$$

for $kT_s \leq t < (k+1)T_s$ (i.e. using zero-order hold). In this controller the differential component is in fact calculated from the $\alpha\beta - tracker$ that acts as a differentiator and low-pass filter (by choosing both α and β smaller than 1). The last term in the above equation is the velocity feed-forward that is added to the control output with static feed-forward gain K_{ff}. As only deviations from the steady-state position error, for constant velocity reference tracking, are important for the print quality, we do not need an integral term in the controller.

When tuning the controller we need to find values for T_s, K_p, K_d, K_{ff}, α and β. The sample frequency ($\frac{1}{T_s}$) of the observer-based controller for the existing implementation in the printer was chosen at 250 Hz, which is about 50 times the required bandwidth of 4 Hz. The sample frequency was chosen this high as classically all controller tuning was done in continuous-time. Discretization by approximating the continuous-time controller was used to implement the controller in discrete-time which requires this high sample frequency. We will use this same sample frequency in combination with the observer-based controller (2)-(5) as a reference for comparison with the event-driven controller, presented in section IV. In section V we will investigate how this observer-based controller performs for a much lower sample frequency of only 62 Hz. This will require some re-tuning of the controller.

The values for the PD controller were chosen at $K_p = 2$, $K_d = 0.3$, $\alpha = 0.75$ and $\beta = 0.25$ such that the relative damping of the system is larger than 0.45 and disturbances are sufficiently rejected at least up to 4.0 Hz. A plot of the sensitivity, that indicates how well the disturbance d is rejected, is depicted in figure 3. In this figure it can be seen that disturbances are rejected up to 4.8 Hz which satisfies the requirement. The feed-forward gain K_{ff} was set to 0.029.

The parameters for the $\alpha\beta - tracker$ were mainly chosen on the basis of simulations. When decreasing α and β, the measurement noise from the measured position and estimated velocity is filtered better, i.e. less "confidence" is expressed in the innovation terms. A disadvantage of lowering the values for α and β is the increased delay in the system. Increasing the delay influences tracking performance but also stability in a negative way.

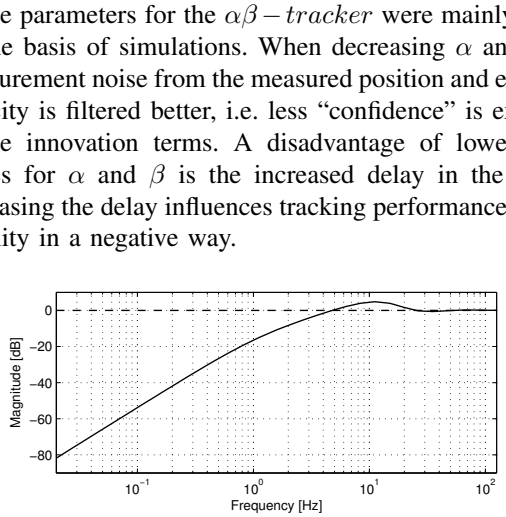

Fig. 3. Sensitivity of the system controlled by the observer-based controller.

IV. EVENT-DRIVEN CONTROL

For the event-driven controller we propose to execute both the measurement and the control update at the moment of a Hall pulse. The control update times are not equidistant in the time domain in this setting, which hampers the use of classical time-driven control schemes. However, we can apply variations of classical design methods, if we define our models of the plant and the controller in the spatial (angular position) domain instead of the time domain, as initially proposed in [11]. This idea is based on the observation that the Hall pulses arrive equally spaced in the spatial domain, as the sensors have an equidistant distribution along the axis of the motor (which is true up to ± 0.2 rad as explained in section II). To use this reasoning, we first transform the motor model (1) to an equivalent model in which the motor angular position is the independent variable. After that we will show how the controller design can be performed using classical control theory.

A. Transformation to spatial domain

The transformation ideas are explained in [11] by the authors. We will recapitulate the main steps of the transformation for this specific example.

The transformation is performed via the following relation:

$$\frac{d\theta}{dt}(t) = \omega(t) \rightarrow \frac{dt}{d\theta}(\theta) = \frac{1}{\omega(\theta)}, \tag{6}$$

where $\omega(\theta)$ denotes the angular velocity of the motor and $t(\theta)$ denotes the time, respectively, at which the motor reaches position θ. Under the assumption that $\omega(t) \neq 0$ for all $t > 0$, a one-to-one correspondence between θ and t exists and an interchange of their roles is possible. Note that $\omega(t) \neq 0$ is valid under normal operating conditions for the considered example, as the motor does not change direction.

Using (6) we obtain the motor model in the spatial domain:

$$\begin{array}{l} \frac{dt}{d\theta}(\theta) = \frac{1}{\omega(\theta)} \\ \frac{d\omega}{d\theta}(\theta) = \frac{1}{J}[-\frac{d(\theta)}{\omega(\theta)} - (\frac{k^2}{R} + B) + \frac{k}{R} \cdot \frac{u(\theta)}{\omega(\theta)}] \\ y(\theta) = t(\theta) \end{array}, \tag{7}$$

where $d(\theta)$ and $u(\theta)$ denote the disturbance torque and the motor voltage, respectively, at motor position θ. Now, time t is a function of θ and is thus a state variable in the new description. Remarkably, to consider the disturbance d as a function of the angular position θ is advantageous for many controller designs, as disturbance are often coupled to the angular position, instead of time. Also in our problem setting (see end of section II), this is the case. Considering disturbances in terms of θ prevents the possibly awkward computation to determine bandwidth requirements in the time domain. Moreover, the output $y(\theta)$ is now the time $t(\theta)$ at which the motor reaches position θ.

The error that is input for the feedback controller is now selected to be the difference between the measured time of a Hall pulse ($t_m(\theta)$) and the time at which the Hall pulse ideally should have occurred based on the reference trajectory (which is denoted by $t_r(\theta)$):

$$e_t(\theta) = t_r(\theta) - t_m(\theta). \tag{8}$$

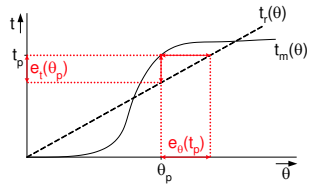

Fig. 4. Errors in t and θ.

This is illustrated in figure 4. In this figure it is also shown how the *time error* can be translated into a *position error*. When ω_r is constant and non-zero in the time interval $(t_r(\theta_p), t_m(\theta_p))$ when $t_m(\theta_p) \geq t_r(\theta_p)$, or $(t_m(\theta_p), t_r(\theta_p))$ when $t_m(\theta_p) < t_r(\theta_p)$, where θ_p is the angular position at an encoder pulse detection, then it holds that

$$e_\theta(t_p) = -\omega_r e_t(\theta_p). \tag{9}$$

When ω_r is not constant, equation (9) can be used as an approximation.

The control objective, that the controller bandwidth should be at least 4.0 Hz, should be translated into a similar requirement in the spatial domain. To define the frequency content of the disturbance, independently of the motor velocity, we analyze the *spatial frequency* $[rad^{-1}]$. The spatial frequency is a characteristic of any structure that is periodic across position in space. It is a measure of how often the structure repeats per unit of distance (completely analogous to "ordinary" frequency with respect to time). The concept of spatial frequency is especially used in wave mechanics and image processing [9]. For the considered disturbance signal, the main component is located at a spatial frequency of $\frac{1}{20 \cdot 2\pi} = 8.0 \cdot 10^{-3}\ rad^{-1}$ (section II). In fact, for the considered example and many other practical examples, most disturbances have a frequency content that varies with the velocity of the controlled actuator. Examples can be found in bearings, axes, rolls, traveling sheets of paper, etc., that all rotate with a velocity that is controlled by the motor. When the motor velocity decreases, all frequencies of the disturbances decrease with the same factor. This might lead to system representations like (7) which might be more convenient in such settings than representations like (1).

B. Controller design

As can be seen from equation (7), the resulting model is non-linear. One way to deal with such a system is by linearizing the model around steady-state trajectories. The steady-state trajectory is chosen straightforwardly at a constant angular velocity ω_e of the motor after start-up:

$$(t_e, \omega_e, d_e, u_e) = (\frac{1}{\omega_e}\theta, \omega_e, d_e, k\omega_e + \frac{(B\omega_e + d_e)R}{k}) \tag{10}$$

where d_e is chosen as the mean value of the disturbance d. The variations around this steady-state trajectory are denoted by $(\Delta t, \Delta\omega, \Delta d, \Delta u)$. Hence, $t = t_e + \Delta t$, $\omega = \omega_e + \Delta\omega$, etc. Around the steady-state trajectory, the linearized dynamics are

$$\begin{array}{l} \frac{d\Delta t}{d\theta}(\theta) = -\frac{1}{\omega_e^2}\Delta\omega(\theta) \\ \frac{d\Delta\omega}{d\theta}(\theta) = \frac{1}{J\omega_e}[-(\frac{k^2}{R} + B)\Delta\omega(\theta) - \Delta d(\theta) + \frac{k}{R} \cdot \Delta u(\theta)] \\ \Delta y(\theta) = \Delta t(\theta) \end{array} \tag{11}$$

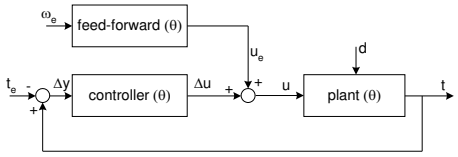

Fig. 5. Feedback/feed-forward controller structure for linearized dynamics.

We can now design a feedback controller that compensates for the first-order variations $\Delta u(\theta)$ as defined in (11). This is schematically depicted in figure 5. The control value applied to the plant should be $u = u_e + \Delta u$. As $u_e = k\omega_e + \frac{B\omega_e R}{k} + \frac{d_e R}{k}$ and we do not know d_e, we selected the feed-forward term so that it takes care of $(k + \frac{BR}{k}) \cdot \omega_e$ and the additional term $\frac{d_e R}{k}$ has to be compensated for by the PD controller. Using the model parameters from section II, we can verify the feed-forward gain that was chosen for the observer-based controller in section III: $K_{ff} = k + \frac{BR}{k} = 0.028 + \frac{3.0 \cdot 10^{-5} \cdot 1.0}{0.028} = 0.029$.

We have chosen to design a PD controller (like in the observer-based controller case) and tuned it in the discrete *position* domain. We aim at using a minimal encoder resolution that satisfies the design objectives (section II). The advantage for choosing a minimal encoder resolution is that the software is interrupted as little as possible to perform control updates, which is beneficial for the processor load. We will investigate if we can achieve sufficient control accuracy with an encoder resolution of only 1 PPR. For this we only need to use one of the three Hall sensors available. Choosing this ultimately low resolution has the advantage that we do not have to deal anymore with inaccurate sensor distributions along the motor axis (of ± 0.2 rad). To tune the controller, equation (11) was first discretized in the spatial domain, using a sample "time" (*sample distance* to be exact) of 2π rad.

The PD controller was chosen as

$$H_{c1}(\hat{z}) = \omega_t \cdot \frac{(K_p + K_d)\hat{z} - K_d}{\hat{z}}, \qquad (12)$$

where the notation $\hat{z}$ is used instead of z to emphasize that the discretization has been made in the spatial domain, instead of the time domain. Furthermore, this event-driven controller takes the time error as input, while the observer-based controller takes the position error as input. To normalize the controller parameters of both controllers, equation (12) is premultiplied by the constant velocity ω_t. The subscript t indicates that the controller is *tuned* for this specific velocity.

To find the values for the controller parameters K_p and K_d, we used the root-locus design method [6]. For $\omega_t = 388$ rad/s and choosing $K_p = 1.0$ and $K_d = 12$ we obtain the root-locus as depicted in figure 6. The '+' marks indicate the roots for feedback gain 1.

To evaluate the disturbance rejection for this controller, we computed the sensitivity function as the transfer function from the disturbance $\Delta d(\theta)$ to $\Delta u_d(\theta)$. Both signals are indicated in figure 7, which shows a graphical representation of equation (11) together with the feedback controller (12).

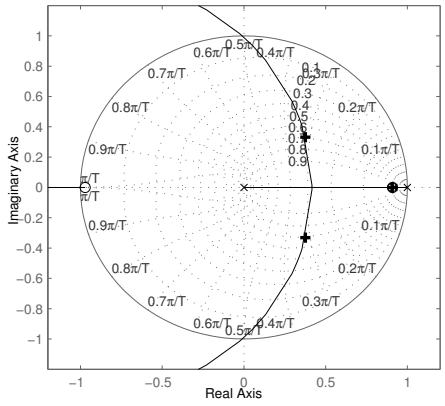

Fig. 6. Root-locus for $\omega_e = 388$; controller (12).

Note that $\hat{s}$ is used instead of s to emphasize that the integration is performed in the spatial domain. The Bode magnitude plot of the discretized position transfer is depicted in figure 8. We see that spatial frequencies up the $0.01\ rad^{-1}$ are attenuated. This satisfies the required bandwidth as given in section IV-A, being $8.0 \cdot 10^{-3}\ rad^{-1}$.

C. Multiple printing speeds

Now we investigated the case in which the printer prints at multiple speeds. For this reason, the system should be analyzed for different values of ω_e. Figure 9 depicts the closed-loop poles for $\omega_e = 200, 300, 400$ and 500 rad/s for the above derived controller. It can be seen that for the various values for ω_e the closed-loop system is behaving differently. For $\omega_e = 200$ rad/s the system is even unstable as the closed-loop poles lie outside the unit circle.

To solve this problem, we propose to schedule the controller gain proportionally with ω_e, i.e. to use a linear parameter-varying (LPV) controller. When the controller is tuned such that it matches (12) for ω_t, we only need to replace ω_t by ω_e in (12), while keeping the same values

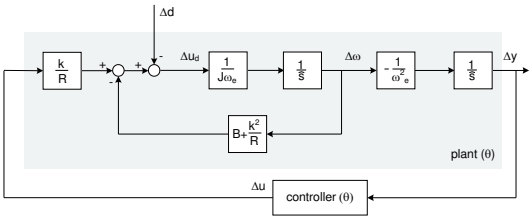

Fig. 7. Graphical representation of linearized dynamics (11).

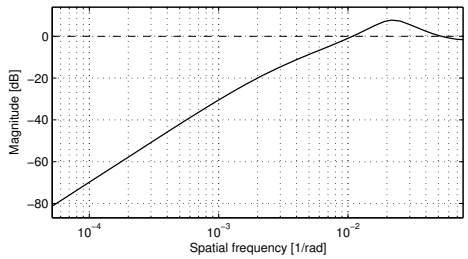

Fig. 8. Bode mag. plot of transfer from $\Delta d(\theta)$ to $\Delta u_d(\theta)$ for $\omega_e = 388$.

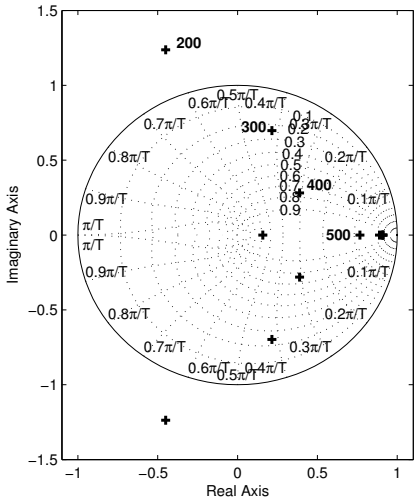

Fig. 9. Pole locations for multiple values of ω_e; controller (12).

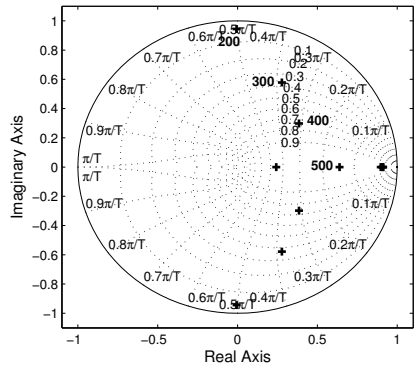

Fig. 10. Pole locations for multiple values of ω_e; controller (13).

for K_p and K_d:

$$H_{c2}(\hat{z}) = \omega_e \cdot \frac{(K_p + K_d)\hat{z} - K_d}{\hat{z}} \tag{13}$$

From the pole locations, as depicted in figure 10, we observe that performance is improved, but for $\omega_e = 200$ rad/s the system is close to instability.

We propose to schedule the controller gains with the aim to keep the closed-loop poles in the spatial domain at approximately the same locations. This is achieved by the controller

$$H_{c3}(\hat{z}) = \frac{\omega_e^2}{\omega_t} \cdot \frac{(K_p + K_d\frac{\omega_e}{\omega_t})\hat{z} - K_d\frac{\omega_e}{\omega_t}}{\hat{z}} \tag{14}$$

The controller gains were again chosen such that the poles for $\omega_e = 388$ rad/s match the poles of controller (12) for $\omega_e = 388$ rad/s.

Figure 11 displays the closed-loop poles for controller (14). This controller results in the approximate same control performance for all evaluated values of ω_e *in the spatial domain*. This can be observed in figure 12 in which the responses to a unit step are depicted for the four evaluated values of ω_e. The interpretation of a unit step here is that the reference *time* instantaneously changes from 0 to 1 second at position $\theta = 0$ rad. Consequently, the step response displays

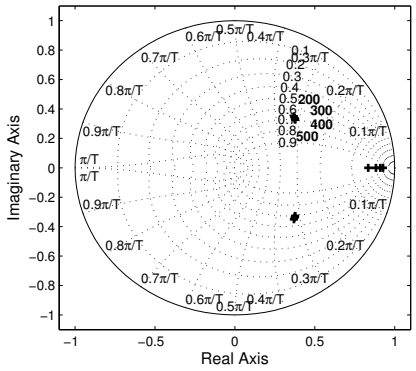

Fig. 11. Pole locations for multiple values of ω_e; controller (14).

the time variation (time error, with respect to the steady state trajectory) when the motor reaches a certain position.

As explained earlier (see equation (9)), we can transform the error to the time domain from $e_t(\theta)$ by means of equation (9), as ω_r is constant after the step has been applied. In the same way we can transform the position scale on the horizontal axis of the step responses to a time scale. The results for the four step responses are plotted in figure 13. It can be seen that the settling *time* decreases as ω_e increases. Hence, the developed controller does not have a (constant) settling *time* but at a (constant) settling *distance*. This makes sense for the printer design, as printing accuracy does not vary with the printing speed in this case.

V. SIMULATION RESULTS

Simulations have been carried out for both the observer-based controller as given in (2)-(5) and the event-driven controller as given in (14), in which a constant reference

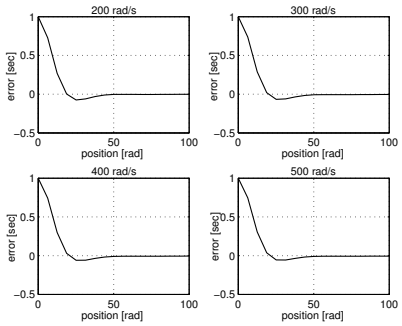

Fig. 12. Step responses for multiple values of ω_e; controller (14).

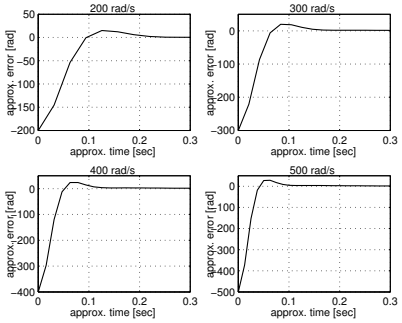

Fig. 13. Step responses for multiple values of ω_e; controller (14).

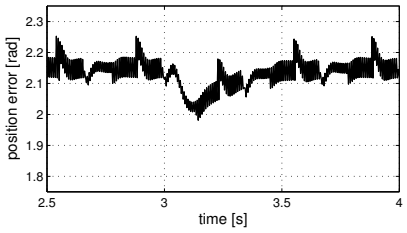

Fig. 14. Simulation results for the observer-based controller (5) with 12 PPR encoder and sample frequency 250 Hz.

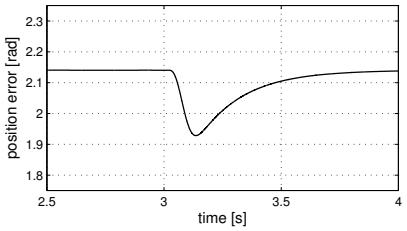

Fig. 15. Simulation results for the event-driven controller (14) with 1 PPR encoder.

velocity of 388 rad/s is tracked. At 3 seconds a disturbance torque pulse d, as depicted in figure 2, is applied to the system. This pulse resembles the disturbance d as induced by a sheet entry at the fuse roll.

The simulation results are depicted in figures 14, 15 and 16, and show the position error for various simulations with the observer-based controller and the event-driven controller proposed in (14) with $e_t(\theta)$ as input.

For ω_e in equation (14) we used an estimation of the actual speed by using the duration from the previous pulse ($t_{p,k-1}$) until the time of the last pulse ($t_{p,k}$), based on the encoder resolution of 2π rad:

$$\omega_e = \frac{2\pi}{t_{p,k} - t_{p,k-1}} \tag{15}$$

From figures 14 and 15 it can be seen that both controllers perform equally well for the case in which the observer-based controller is running at 250 Hz and 12 PPR and the event-driven controller running at 1 PPR (resulting in an average sample frequency of 62 Hz). Both position errors vary within a range of about ± 0.2 rad. The main difference is that the error is smooth in the event-driven controller situation, but a high frequent ripple is visible in the observer-based situation. This is caused by the small errors that are modeled as inaccuracies in the distribution of Hall pulses over 1 rotation of the motor.

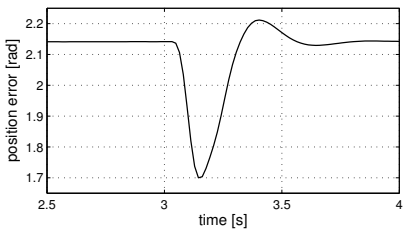

Fig. 16. Simulation results for the observer-based controller (5) with 1 PPR encoder and sample frequency 62 Hz.

Figure 16 shows the position error for the same situation, but here an observer-based controller in combination with a 1 PPR encoder, running at 62 Hz, was used. This frequency was specifically chosen as the event-driven controller also runs at an average frequency of 62 Hz. For this configuration the controller had to be re-tuned, to still obtain a bandwidth of 4 Hz. The following parameter values were obtained: $T_s = \frac{1}{62}s, K_p = 1, K_d = 0.05, \alpha = 1$ and $\beta = 1$. Because we only have 1 PPR, the inaccurate position information caused by the distribution of the Hall pulses has disappeared. For this reason both parameters of the $\alpha\beta - tracker$ could be set to one, i.e. no filtering was necessary (as can be seen from equation (3)). The value for K_d needed to be lowered considerably as otherwise the computed actuator signal would be impossible to realize with the motor amplifier. Comparing figures 15 and 16 we observe that the event-driven controller outperforms the observer-based controller in this setting, in the sense that the error deviation from the constant equilibrium value is 0.2 rad for the event-driven controller and 0.5 rad for the observer-based controller, which is out of spec.

VI. MEASUREMENT RESULTS

To validate the simulation results of the previous section, we compared the observer-based controller (2)-(5) with the event-driven controller as proposed in (14) by implementing them both on a complete prototype document printing system at Océ technologies BV.

The model used in section II was already matched with this prototype system. Therefore, the controller parameters obtained from the analysis and synthesis described in section IV-B could be applied *directly* to control the TTF belt in the prototype.

The experimental results for both controllers are given in figures 17 and 18. These figures show the position error during printing over 5 seconds (after start-up). In this period, 5 sheets are printed at a speed of 80 pages per minute. As the control performance was measured with the position error in rad, the results can be compared with the simulation results (section V).

Comparing the results in figures 17 and 18 we observe, as expected from the simulations, similar control performance for the observer-based controller and the event-driven controller, both within spec. The maximum deviation from the average position error is for both controllers about 0.15 rad, which is smaller than 0.25 rad as required (see section II). However, keep in mind that the event-driven controller uses an encoder with a resolution that is a factor 12 lower than for the observer-based controller. Furthermore, the observer-based controller runs at a (constant) control sample frequency of 250 Hz and the event-driven controller at a much lower *average* frequency (approximately 62 Hz). The errors caused by the sheet passings can be distinguished in both figures, although there are more disturbances (at different frequencies) acting on the system as can be seen from the measurement data.

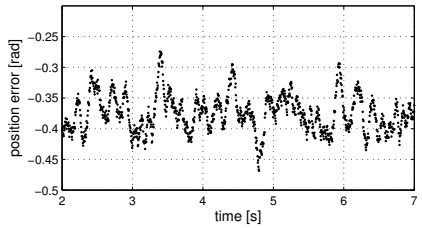

Fig. 17. Experiment observer-based controller (5) with 12 PPR encoder and sample freq. 250 Hz.

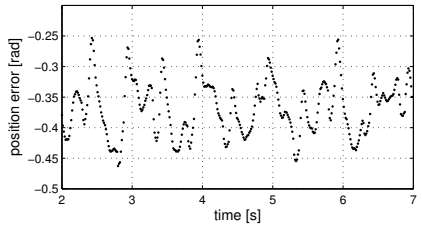

Fig. 18. Experiment event-driven controller (14) with 1 PPR encoder.

VII. PROCESSOR LOAD

Next to the cost price reduction by using cheap encoders, another important reason to implement the proposed event-driven controller is to reduce the processor load. Therefore, we measured the processor load during the experiments. As a measure for the processor load we used the time the processor needs to execute the controller computations over the 5 seconds experiments. For the event-driven controller, this resulted in a processor load of 30 ms. This value can also be obtained by multiplying the computation time of one control update, which is approximately 98 μs, with the measurement time (5 seconds) and the average sample frequency of 62 Hz. The processor load of the observer-based controller was measured at 130 ms. This value is equal to 104 μs per control update multiplied by the measurement time and the 250 Hz sample frequency. From these numbers we obtain that the event-driven controller reduces the processor load for the controller task with a factor 4.3, compared to the originally implemented observer-based controller.

VIII. CONCLUSIONS

In this paper we presented the use of event-driven control to accurately control a brushless DC-motor in a high speed printer on the basis of a Hall encoder having a resolution of 1 pulse per revolution. By means of analysis, simulation and industrial experiments we showed that the performance of the controller satisfied the requirements. Furthermore, we showed that similar control performance could be achieved as the initially proposed industrial observer-based controller. However, the event-driven controller used an encoder with a resolution 12 times lower and was running at a much lower average sample frequency, and therefore involved a significantly lower processor load. Compared to the originally implemented observer-based controller implementation, a processor load reduction could be obtained of a factor 4.3 with the event-driven controller. The advantages of the event-driven controller over the observer-based controller can be summarized as follows:

- Only a cheap encoder with a resolution of 1 PPR necessary, instead of 12 PPR;
- A lower computational load for the processor;
- Less tuning parameters for control algorithm.

The analysis and controller synthesis were based on the observation that the controller triggering is synchronous in the spatial domain. We were able to write the control problem, that was asynchronous in the time domain, as a synchronous problem in the spatial domain. With this, we were able to apply conventional control theory to design and tune the controller. The resulting control performance, as obtained from the analysis, is defined in the spatial domain. When disturbances are also acting in the spatial domain, which is often the case, we can easily model them in the spatial domain and determine how they are rejected. Furthermore, we can now use the settling *distance* as a control performance measure, instead of the settling *time*. In many cases, also tracking requirements are better defined in the spatial domain.

REFERENCES

[1] Årzén, Karl-Erik (1999). A simple event-based PID controller. In: *Proceedings of the 14th World Congress of IFAC*. Beijing, P.R. China, Vol. 18, pp. 423–428.

[2] Åström, K.J., and B.M. Bernhardsson (2002). Comparison of Riemann and Lebesgue sampling for first order stochastic systems. In: *Proceedings of the 41st IEEE Conference on Decision and Control*, Las Vegas, Nevada, USA, pp. 2011–2016.

[3] Blackman, S.S. (1986). Multiple target tracking with radar applications. Norwood, MA: Artech House.

[4] Bode, H. (1945). Network Analysis and Feedback Amplifier Design. New York: Van Nostrand Reinhold.

[5] Bruin, D. de and P.P.J. van den Bosch (1998). Measurement of the lateral vehicle position with permanent magnets. In: *Proc. of IFAC workshop on Intelligent Components for Vehicles*, Spain, pp. 9–14.

[6] Franklin, G.F., J.D. Powell and M.L. Workman (1998). Digital control of dynamic systems. *Third edition, MA: Addison-Wesley.*

[7] Friedland, B. (1973). Optimum steady-state position and velocity estimation using sampled position data. In: *IEEE transactions on Aerospace and Electronic Systems*. Vol. AES-9, No. 6, pp. 906–911.

[8] Glad, T. and L. Ljung (1984). Velocity estimation from irregular, noisy position measurements. In: *Proceedings of the IFAC 9th Triennial World Congress*, Budapest, No. 2, pp. 1069–1073.

[9] Goodman, J. (2005). Inroduction to Fourier optics. 3rd ed., Robers & Company, 2005.

[10] Grewal, M.S. and A.P. Andrews (1993). Kalman filtering: theory and practice. Englewood Cliffs: Prentice Hall.

[11] Heemels, W.P.M.H., R.J.A. Gorter, A. van Zijl, P.P.J. van den Bosch, S. Weiland, W.H.A. Hendrix and M.R. Vonder (1999). Asynchronous measurement and control: a case study on motor synchronization. In: Control Engineering Practice, Vol. 7, pp. 1467–1482.

[12] Kofman, E., and J.H. Braslavsky (2006). Level crossing sampling in feedback stabilization under data-rate constraints. In: *Proceedings of the 45th IEEE Conference on Decision and Control*, San Diego, California, USA, pp. 4423–4428.

[13] Krucinski, M., Cloet, C., Tomizuka, M. and R. Horowitz (1998). Asynchronous observer for a copier paper path. In: *Proceedings of the 37th IEEE Conference on Decision and Control*, Tampa, Florida, Vol. 3, pp. 2611–12.

[14] Phillips, A.M., and M. Tomizuka, (1995). Multirate estimation and control under time-varying data sampling with applications to information storage devices. In: *Proceedings of the 1995 American control conference*, Vol. 6, pp. 4151–4155.

[15] Vottis, C.V. (2003), Extracting more accurate position and velocity information from optical incremental encoders, SAI/2yr Thesis, Technische Universiteit Eindhoven, ISBN 90-444-0335-4.